Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000), Breckenridge, Colorado, USA 12-15 April 2000

Anytime Diagnostic Reasoning using Approximate Boolean Constraint Propagation

Alan Verberne*
Dept. of AI, Faculty of Sciences
Vrije Universiteit Amsterdam
a.s.verberne@research.kpn.com

Frank van Harmelen
Dept. of AI, Faculty of Sciences
Vrije Universiteit Amsterdam
Frank.van.Harmelen@cs.vu.nl

Annette ten Teije
Dept. of CS
University of Utrecht
annette@cs.uu.nl

Abstract

In contrast with classical reasoning, where a solution is either correct or incorrect, *approximate reasoning* tries to compute solutions which are close to the ideal solution, without necessarily being perfect. Such approximate reasoning can be used to exchange solution quality for computation time, known as anytime reasoning.

In this paper we study approximate versions of *diagnostic reasoning*. Traditionally, diagnostic reasoning is characterised in terms of the logical entailment relation. In this paper we study the effects of replacing the logical entailment relation with an approximate version of the entailment relation, in particular an approximate version of Boolean Constraint Propagation (BCP).

We characterise the cheapest versions of approximate BCP which allows single components and entire systems to be diagnosed correctly. From these upperbounds surprisingly low values follow which are needed to correctly diagnose many of the typical circuit examples from the literature. A particularly interesting property that we discovered is that the point at which approximate diagnosis coincides with classical diagnosis is entirely determined by the nature of the individual components, and not by either the size or the structural complexity of the overall device.

Keywords: Deduction, Diagnosis

*currently at KPN Research

1 MOTIVATION

Approximate reasoning is a form of reasoning that can be used for solving complex problems. For approximate reasoning one defines a quality measure on the output, and the computation then tries to optimize this quality measure. This is in contrast with the conventional notion of a solution, where a solution is either correct or incorrect, with no middle ground. Optimising the quality measure does provide such a middle ground.

Approximate reasoning is interesting for several reasons. First of all, most AI-problems (tasks) are hard problems in term of their complexity measure. Planning, diagnosis and configuration are all examples of AI tasks for which even simple varieties are already intractable (e.g. [Bylander *et al.*, 1991]). Therefore it is necessary to look for cheaper but approximate solutions instead of intractable precise solutions. Secondly, it depends on the particular application of the problem type (e.g. design, diagnosis) whether a precise solution is actually needed or whether an approximate solution suffices. For instance, in diagnostic reasoning there is not always a need for computing precise diagnoses, for example in cases where all possible diagnoses will result in the same repair action (e.g. when all fault-candidates are located on the same computer-board that must be replaced or in the medical domain when all possible diagnoses indicate the same drug to be prescribed) [vanHarmelen & tenTeije, 1995]. As a third reason, it is often not possible in practice to have complete and correct data and knowledge. Examples are missing attribute values in classification, incomplete medical knowledge for performing diagnosis and incomplete requirements for design. So again, an approximate answer should be computed if possible, because an approximate answer is often better than no answer at all.

A particularly interesting form of approximate reasoning is *anytime reasoning* [Dean & Boddy, 1988; Boddy & Dean, 1989; Zilberstein & Russell, 1996]. The most important characteristic of anytime reasoning is that with increasing runtime, the quality of the solution increases. Furthermore, the reasoning can be interrupted at any time and will return the best result computed until then. In this paper, we will concentrate on this type of reasoning.

Many types of reasoning in AI such as diagnosis, classification, design and planning are characterised in terms of logical entailment [Reiter, 1987; Green, 1969; Rosenschein, 1981]. A general approach for constructing approximate problem solving behaviour is to use an approximation of the logical entailment for characterising such a problem type and see what conclusions can be drawn using approximate reasoning for a particular problem. This approach was already taken in [tenTeije & vanHarmelen, 1996], where we used the approximate entailment relations defined in [Schaerf & Cadoli, 1995] to characterise approximate diagnostic reasoning. In this paper we consider the sound but incomplete approximation of the logical entailment from [Dalal, 1996] in the context of diagnostic reasoning. The approximation of Dalal is called BCP_k and is based on *boolean constraint propagation*. The parameter k can be increased to improve the quality (and of course also the cost) of this approximate entailment relation as will be explained in section 3.

The general question that we study is *"how can we use BCP_k for approximate diagnostic reasoning?"*. This question boils down to questions such as "how does the quality of the diagnoses improve with time using BCP_k?", "for which k do the BCP_k diagnoses coincide with the classical diagnoses?", "what can be said about the incorrect answers that BCP_k gives?", etc.

The structure of the paper is as follows: to make this paper self-contained we repeat briefly in section 2 the standard definitions of consistency-based diagnosis from the literature. Section 3 introduces the approximate entailment relation we will exploit for diagnostic reasoning. The main results of this paper are discussed in section 4, where we present examples and theorems that characterise the behaviour of the approximate diagnostic reasoning that results from applying BCP_k in the standard definition of diagnosis. Finally, section 6 summarises and concludes.

2 CONSISTENCY-BASED DIAGNOSIS

We briefly repeat the widely accepted definitions from [Reiter, 1987]:

Definition 1 (From [Reiter, 1987])

- A *system* is a pair $(\mathrm{SD}, \mathrm{COMP})$ where SD, the *system description* is a set of sentences in first order logic, and COMP the set of system components is a set of constants. The system description uses abnormality predicates $ab(c)$ for every $c \in \mathrm{COMP}$, interpreted to mean that component c does not function normally.
- The *observations* are a set of first order sentences OBS.
- A *diagnosis problem* exists *iff* $\mathrm{SD} \cup \mathrm{OBS} \cup \{\neg ab(c) | c \in \mathrm{COMP}\}$ is inconsistent.
- A *candidate-set* is set $\Delta \subseteq \mathrm{COMP}$.
- A *diagnosis* is a non-empty candidate-set Δ such that $\mathrm{SD} \cup \mathrm{OBS} \cup \{\neg ab(c) | c \in \mathrm{COMP} \setminus \Delta\}$ is consistent.
- A *minimal diagnosis* is a set Δ such that Δ is a diagnosis but no $\Delta' \subset \Delta$ is a diagnosis.
- A *conflict set* is a set $C \subseteq \mathrm{COMP}$ such that $\mathrm{SD} \cup \mathrm{OBS} \cup \{\neg ab(c) | c \in C\}$ is inconsistent.

Although in general, SD can be an arbitrary set of first-order sentences, in practice (and also in our approach), SD is assumed to consist of two parts: the *behaviour model* which describes the normal behaviour of each individual component, and the *structure model* which describes the connections between the individual components.

A slight complication arises because the approximate deduction relation BCP_k that we intend to use is only defined for propositional theories. Fortunately, typical behaviour and structure models are in fact function-free and range over finite alphabets. Such first-order theories can be rewritten as equivalent propositional theories using standard methods.

3 DEFINING $\vdash_k^{BCP}$

This section repeats some of the definitions and results from [Dalal, 1996] and [Dalal & Yang, 1997]. They define the family of approximate deduction relations BCP_k.

Boolean Constraint Propagation [McAllester, 1990] is a limited form of deduction which is a variant of unit resolution [Chang & Lee, 1973]. It performs limited deduction in linear time, as follows: given a theory T[1], BCP monotonically expands T by adding literals as follows: in each step, if any single clause in T and all

[1] We will assume all theories to be propositional, and in clausal normal form.

the literals in T taken together entail any other literal (or $\perp$), then this literal (or $\perp$) is added to the theory T. [Dalal, 1996] defines BCP algebraically:

Definition 2 ($\vdash_{BCP}$) *A formula ϕ is inferable from a theory T using BCP, denoted by $T \vdash_{BCP} \phi$, iff $T \cup \{\neg\phi\} =_{BCP} \perp$ via the following rewrite rules for $=_{BCP}$:*

1. $\{\perp\} \cup T \quad =_{BCP} \quad \{\perp\}$
2. $\{(\alpha), (\neg\alpha \vee \alpha_1 \vee ... \vee \alpha_n)\} \cup T \quad =_{BCP}$ $\{(\alpha), (\alpha_1 \vee ... \vee \alpha_n)\} \cup T$

where α and the α_i's are any literals.

BCP is a sound but incomplete (ie approximate) deduction relation:

Example 1

$\{(P \vee Q), (P \vee \neg Q), (\neg P \vee Q), (\neg P \vee \neg Q)\} \vdash \perp$
$\{(P \vee Q), (P \vee \neg Q), (\neg P \vee Q), (\neg P \vee \neg Q)\} \not\vdash_{BCP} \perp$

This shows that $\vdash_{BCP}$ cannot chain on the intermediate result P. Although incomplete in general, BCP is complete for Horn theories. Its incompleteness comes from the inability to use previously inferred clauses during the reasoning process (ie "chaining"):

Example 2 *Let $T_0 = \{(P \vee Q), (P \vee \neg Q), (\neg P \vee S \vee T), (\neg P \vee S \vee \neg T)\}$, then*

- *$T_0 \vdash_{BCP} P$ (by adding $\neg P$ to T_0 and deriving $\perp$ using the first two clauses), and*
- *$T_0 \cup \{P\} \vdash_{BCP} S$ (by adding $\neg S$ and using the last two clauses), but*
- *$T_0 \not\vdash_{BCP} S$.*

Allowing BCP to chain on arbitrary clauses would make it sound and complete and would therefore render it uninteresting as an approximate deduction method.

This is the motivation for defining BCP$_k$ [Dalal, 1996], where chaining is allowed, but only on formulae of limited length:

Definition 3 ($\vdash_k^{BCP}$) *For any theory T and clauses ϕ and ψ:*

$$\frac{T \vdash_{BCP} \phi}{T \vdash_k^{BCP} \phi}, \quad \frac{T \vdash_k^{BCP} \psi;\ (T, \psi) \vdash_k^{BCP} \phi}{T \vdash_k^{BCP} \phi} \ \textit{if } |\psi| \leq k$$

Clearly, when k increases, $\vdash_k^{BCP}$ is allowed to chain on ever more formulae, and will become ever more complete. For example, $T_0 \vdash_0^{BCP} P$, $T_0 \not\vdash_0^{BCP} S$ and $T_0 \vdash_1^{BCP} S$. $\vdash_1^{BCP}$ is complete for 2-CNF theories.

4 USING BCP$_k$ FOR DIAGNOSIS

This section contains our results of applying BCP$_k$ to diagnostic reasoning. First we will start with the effect of using a sound but incomplete reasoner for Reiter's definition of diagnostic reasoning. After this, we give a number of examples of diagnostic reasoning using BCP$_k$ as illustration. We continue with some intuitions of using BCP$_k$ in diagnosis and finally we present our results of anytime diagnostic reasoning using approximate BCP.

4.1 USE OF A SOUND, BUT INCOMPLETE REASONER

Using a sound but incomplete approximation of the entailment relation (i.e. BCP$_k$ or the 3-S-approximation from [Schaerf & Cadoli, 1995]) in Reiter's definition of diagnosis has two effects. First, using such an approximation results in less diagnostic problems. The assumption that all components are functioning correctly, together with the observed behaviour and the system description could be classically inconsistent but still consistent under the approximate entailment relation (since $\perp$ could be deducible classically, but not by the weaker approximate entailment):

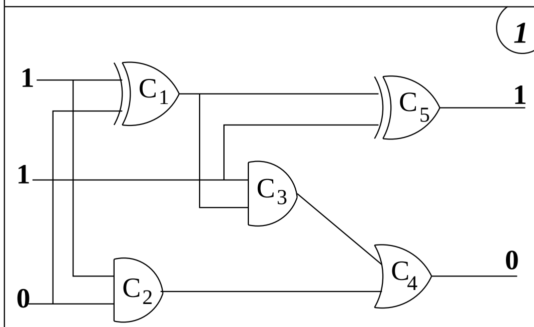
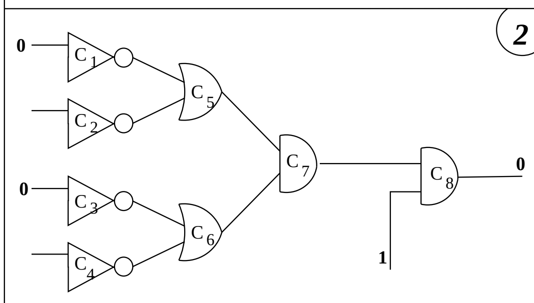
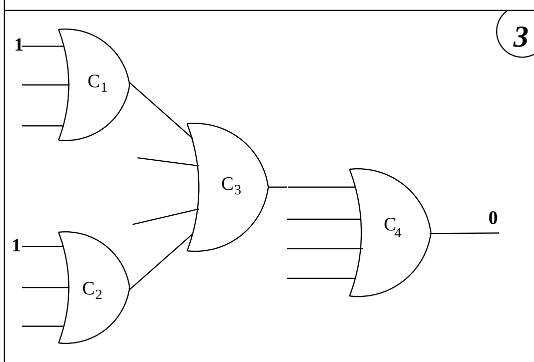
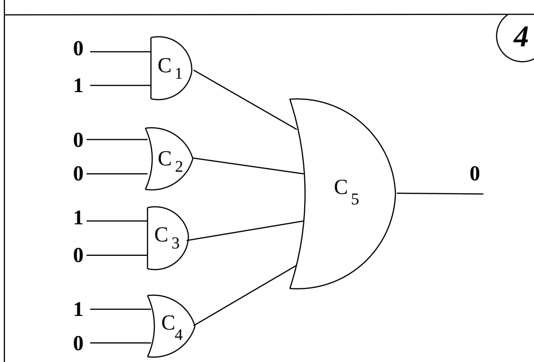
Theorem 1 (Approximate diagnostic problems are classical problems)
For any sound but incomplete entailment relation $\mid\!\sim$:
if $\quad SD \cup OBS \cup \{\neg ab(c) | c \in COMP\} \mid\!\sim \perp$
then $SD \cup OBS \cup \{\neg ab(c) | c \in COMP\} \vdash \perp$

The second effect is that diagnoses which are not classical diagnoses are possibly approximate diagnoses. In the classical case such a diagnosis together with the observed behaviour and system description derives $\perp$. Again, because of the incompleteness of the approximation, $\perp$ might not follow in the approximate case and therefore it could be an approximate diagnosis:

Theorem 2 (Every classical diagnosis is an approximate diagnosis)
For any sound but incomplete approximate entailment relation

$$\{\Delta | \Delta \textit{ is a classical diagnosis}\} \subseteq$$
$$\{\Delta | \Delta \textit{ is an approximate diagnosis}\}$$

NB: This result only holds if we consider the set of *all* diagnoses. It is no longer true when restricted to the set of minimal diagnoses.

System	value of k	Correct minimal k-diagnoses
 (From [Reiter, 1987], Figure 1)	$k >= 0$	{c1}, {c3,c5}, {c4,c5}
 (From [Davis, 1988], Figure 20)	$k = 0$	{}
	$k >= 1$	$\{c_1\}, \{c_3\}, \{c_5\}, \{c_6\}, \{c_7\}, \{c_8\}$
	$k = 0, 1$	{}
	$k = 2$	$\{c_1\}, \{c_2\}, \{c_3\}, \{c_4\}$
	$k >= 3$	$\{c_1, c_2\}, \{c_3\}, \{c_4\}$
	$k = 0$	$\{c_4\}, \{c_5\}, \{c_1, c_2\}, \{c_1, c_3\}, \{c_2, c_3\}$
	$k = 1$	$\{c_4\}, \{c_5\}, \{c_1, c_2, c_3\}$
	$k >= 2$	$\{c_4\}, \{c_5\}$

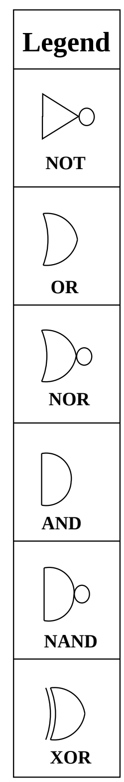


Figure 1: Examples of approximate reasoning in diagnosis using $\vdash_k^{BCP}$

4.2 EXAMPLES AND INTUITION

Figure 1 describes a number of examples of how BCP$_k$ acts in diagnosis. The left-hand side gives the system descriptions and the observations. On the right-hand side the resulting diagnoses are given together with the

k for which they were discovered. Below we describe how the k parameter influences the outcome of the reasoning process.

Combining the definition of diagnosis from [Reiter, 1987] and the definitions of BCP_k allows us to define the notion of a *BCP_k diagnosis*:

Definition 4 (BCP_k diagnosis)
A non-empty candidate diagnosis set $\Delta \subseteq \text{COMP}$ is a *BCP_k diagnosis* iff
$\text{SD} \cup \text{OBS} \cup \{\neg ab(c) | c \in \text{COMP} \setminus \Delta\}$ is BCP_k-consistent.

4.2.1 BCP_0 for a Single Component

The k parameter determines the quality of the computed diagnosis. Its influence is illustrated by means of a simple example. Consider performing diagnosis on a system consisting of only one AND gate in figure 2.

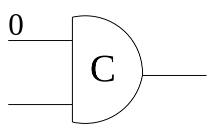


Figure 2: An AND gate with one known input

The system description only contains one behaviour model, and no structural model. The behaviour model of c is[2]:

$$\begin{array}{l} ab\_c \vee \neg in_1\_c\_0 \vee \neg in_2\_c\_0 \vee out\_c\_0 \\ ab\_c \vee \neg in_1\_c\_0 \vee \neg in_2\_c\_1 \vee out\_c\_0 \\ ab\_c \vee \neg in_1\_c\_1 \vee \neg in_2\_c\_0 \vee out\_c\_0 \\ ab\_c \vee \neg in_1\_c\_1 \vee \neg in_2\_c\_1 \vee out\_c\_1 \end{array}$$

The first of these states that given two 0 input values and assuming correct behaviour, the output also equals 0 and similar for the other formulae.

Assume that we are given the observation $\text{OBS} = \{in_1\_c\_0\}$ and that we want to derive that the output of this component is 0: out_c_0, assuming it is working correctly: $\neg ab\_c$.

$$SD \cup \{in_1\_c\_0\} \cup \{\neg ab_c\} \vdash_0^{BCP} out\_c\_0$$

Adding $\neg out\_c\_0$ to the theory, and performing unit resolution on these literals and the behaviour model results, among others, in:

$$\begin{array}{l} \neg in_2\_c\_0 \\ \neg in_2\_c\_1 \end{array}$$

[2]Notations such as out_c_0 are the propositional translation of the first-order formula $out(c) = 0$.

As a result of the translation into propositional logic, the clause $in_2\_c\_1 \vee in_2\_c\_0$ is in SD. This can be used to derive the empty clause using the previous two literals, establishing the desired conclusion. This shows that $k = 0$ allows some inference over the behaviour of components.

4.2.2 Connections Between Components

To allow computed values to spread throughout an entire system, they have to be propagated from one component to another.

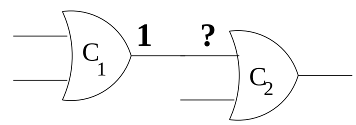


Figure 3: Propagating values from one component to another

Consider the example in figure 3. An output value of 1 is known for component c_1. To propagate this value towards c_2, the value has to go through the link connecting these two components. After propositional translation, the formulas in the structure model that describe the link in the above example look like (For all v in some defined set VALUES):

$$\neg out\_c_1\_v \vee in_1\_c_2\_v \text{ and } \neg in_1\_c_2\_v \vee out\_c_1\_v$$

If for example $out\_c_1\_1$ is known, then also its counterpart, $in_1\_c_2\_1$ is BCP inferable. In other words, if a value assignment to one side of a link can be inferred for a certain k, the value on the other side of the link can be inferred as well.

4.2.3 BCP_0 for Multiple Components

Consider the following example:

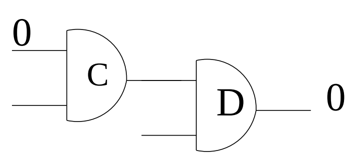


Figure 4: Reasoning about multiple components

As before, we try to use BCP_0 to derive that the output of this circuit is 0:

$$SD \cup \{in_1\_c\_0\} \cup \{\neg ab\_c, \neg ab\_d\} \vdash_0^{BCP} out\_d\_0$$

The SD for this circuit consists of clauses for both and-gates, augmented with the clause stating the connection between the two. Unlike the single-component case from section 4.2.1, no new literals can be inferred. The only possible steps yield the following binary clauses (resulting from the description of C):

$$\neg in_2\_c\_0 \vee out\_c\_0$$
$$\neg in_2\_c\_1 \vee out\_c\_0$$

This states that regardless of its second input value, C's output will be 0. Although this is classically derivable, such a combination of two binary clauses is not allowd under BCP_0. As a result, BCP_0 is not able to show that the output of the entire circuit is 0 either.

Observation: Notice that BCP_0 would not have failed on this example if either of the other two input values had been given. Classically, these values are irrelevant to the output of the circuit. For BCP_0 however, this redundant information would have been just enough to achieve the desired conclusion. For example, knowing either $in_2\_c\_0$ or $in_2\_c\_1$ would allow further reduction of one of the above two binary clauses, leading to out_c_0.

4.2.4 BCP_1 for Multiple Components

Now we will show that unlike BCP_0, BCP_1 is able to derive the output of the above example:

$$SD \cup \{in_1\_c\_0\} \cup \{\neg ab\_c, \neg ab\_d\} \vdash_1^{BCP} out\_d\_0 \quad (1)$$

According to definition 3, we are allowed to hypothesise a formula of length k (here $k = 1$, ie a literal), to prove this hypothesis, and then exploit this result in the subsequent reasoning. Such chaining on formulae of length 1 is exactly what was disallowed under BCP_0. In the example of fig. 4 the obvious literal to use is out_c_0. This is easily established under BCP_1 (in fact, we saw it already followed under BCP_0). We are then allowed to add out_c_0 to the left hand side of (1). This enables a similar inference on component D to derive the required result.

4.2.5 BCP_k for more complex components

Notice that the circuit from fig. 4 is equivalent to a single AND-component with three input gates: The

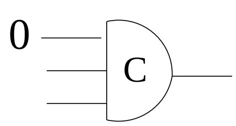


Figure 5: Reasoning about multiple components

same results as above hold for this complex component: BCP_0 cannot derive out_c_0, while BCP_1 can. BCP_0 cannot further reduce the following set of binary clauses:

$$\neg in_2\_c\_1 \vee \neg in_3\_c\_1$$
$$\neg in_2\_c\_1 \vee \neg in_3\_c\_0$$
$$\neg in_2\_c\_0 \vee \neg in_3\_c\_1$$
$$\neg in_2\_c\_0 \vee \neg in_3\_c\_0$$

As before, classically this set of clauses is inconsistent (as desired), but BCP_0 cannot derive this. BCP_1 is able to derive this inconsistency by first hypothesising (for instance) $in_2\_c\_0$.[3] This hypothesis can be proved using the clause $in_2\_c\_0 \vee in_2\_c\_1$ (resulting from the translation to propositional logic) and the same clause for in_3_c, in combination with the first two binary clauses above. Now we can chain on this result ($in_2\_c\_0$) which gives the required empty clause by combining the last two binary clauses above with the exclusion clause for in_3_c.

Summarising: although BCP_0 can be used to reason about single simple components, and about the connections between them, higher values of k are required to reason about more complex circuits. These intuitions will be made more precise in the subsequent theorems.

5 THEOREMS ON DIAGNOSTIC REASONING WITH BCP_k

After presenting the intuition of using BCP_k, this section presents the formal results about the effects of using BCP_k in diagnostic reasoning. After some initial and easy results, we will be mostly concerned with upperbounds on values for k: for which values of k do classical diagnosis and approximate diagnosis coincide. Some of these results are quite surprising and yield much lower values for k than might be expected.

5.1 EFFECTS OF THE INCOMPLETENESS OF BCP_k

In section 4.1, we mentioned the general effect of a sound and incomplete approximation of the entailment relation for consistency based diagnosis. As a special case, the following holds for BCP_k:

Theorem 3 (Every k diagnostic problem is also a $k+1$ diagnostic problem)
if $SD \cup OBS \cup \{\neg ab(c) | c \in COMP\}$ is BCP_k-inconsistent then $SD \cup OBS \cup \{\neg ab(c) | c \in COMP\}$ is BCP_{k+1}-inconsistent

[3] In fact, any of the literals in these clauses would have done the trick.

Theorem 4 (Increasing k leads to fewer diagnoses)
For any SD, COMP and OBS:

$$\{\Delta|\Delta \textit{ is a } BCP_{k+1} \textit{ diagnosis}\} \subseteq \{\Delta|\Delta \textit{ is a } BCP_k \textit{ diagnosis}\}$$

As already stated after theorem 2, this is only true for the set of all diagnoses. Examples 3 and 4 from figure 1 show that this result does not hold when restricted to minimal diagnoses. For example, in example 3 from the figure, $\{c_1, c_3\}$ is a diagnosis at $k = 2$ as well, albeit not a minimal not.

5.2 AN UPPERBOUND FOR k ON SINGLE COMPONENTS

In general, the values that BCP_k can derive depend on the value of k (obviously), the complexity of the component (section 4.2.5) and the set of known input or output values of the component (the final observation from section 4.2.1). The following lemma provides a precise characterization of the relation between the value of k and the formulas that can be inferred about a single component.

Lemma 1 (Upperbound for k for a single component)
Consider a system $(SD, COMP)$, a set VALUES and an observation set OBS. For every component $c \in COMP$ having n links going in or out, of which $m(\leq n)$ values are elements of OBS and a literal ϕ, which is either $\perp$ or an assignment of a value $v \in VALUES$ to an input or output of c, the following holds:

If $k \geq n - m - 1$ and $SD \cup OBS \cup \{\neg ab\_c\} \vdash \phi$, then $SD \cup OBS \cup \{\neg ab\_c\} \vdash_k^{BCP} \phi$

Next, we will use this lemma to establish an upperbound for the value of k that is required to reason about entire circuits instead of only single components.

5.3 AN OBSERVATION-INDEPENDENT UPPERBOUND for k

It is possible to derive an upper bound of k for a given system (SD,COMP) that is needed to compute classical diagnoses for an arbitrary diagnostic problem. A surprising result is that this upperbound depends only on the types of the components in COMP, and not and not on how they are connected. This upperbound is computed as follows:

First we apply lemma 1 to each component with $m = 0$ (no observations, ie. the worst case), giving as value $n - 1$ for each component, where n is the number of links going in or out. We then take the maximum of all these values:

Definition 5 *$k_{max}(SD, COMP)$ = maximum of $n - 1$ over all components in COMP*

Theorem 5 (Invariance of k with system size)
Let SD and SD′ be two system descriptions which contain the same types of components (ie they only differ in the number of components and how these components are connected, their structure model), then

$$k_{max}(SD) = k_{max}(SD')$$

Proof sketch: Lemma 1 ensures that a component can be correctly simulated at $k = n - 1$ (taking $m = 0$, the worse case). An entire circuit is simply the union of the component descriptions (we can ignore the connection clauses, since section 4.2.2 showed that the connection clauses can be dealt with even by $k = 0$). The k required for an entire circuit is therefore not higher than the k for the most complex component, ie. the maximum of $n - 1$ over all components. □

Lemma 1 ensures that for this value of k we will obtain only the classical diagnoses.

5.4 AN OBSERVATION-DEPENDENT UPPERBOUND for k

Although the previous result stated that the maximally needed value of k is independent of the system interconnection, this value can in general be lowered when information about the given observations is exploited. We will now derive the maximal value for k that is needed for a given SD and a set of observations OBS. For this value of k or higher, BCP_k diagnoses coincide with the classical results.

The characterisation of the upper bound of k is based on conflict sets.

Definition 6
• *Given SD, OBS and a minimal conflict set mcs, ie. $SD \cup OBS \cup \{\neg ab(c)|c \in mcs\} \vdash \perp$, then $k_{needed}(mcs, SD, OBS)$ is defined as the minimal k for which $SD \cup OBS \cup \{\neg ab(c)|c \in mcs\} \vdash_k^{BCP} \perp$;*
• *Given all minimal conflict sets $mcs_1 \ldots mcs_n$ for a diagnostic problem (SD,COMP,OBS), k^* is defined as follows:*

$$k^* = max_{1 \leq i \leq n}(k_{needed}(mcs_i, SD, OBS))$$

This definition of k^* allows the following theorem:

Theorem 6 (When $\mathbf{BCP_k}$ diagnoses equal classical diagnoses)

Δ *is a* BCP_{k^*} *diagnosis for* $(SD, COMP, OBS)$ iff Δ *is a classical diagnosis for* $(SD, COMP, OBS)$.

Theorem 6 is of great importance since it provides an upper bound for k. No k greater than k^* needs to be tried, since the results can not get any better.

The question remains how the minimal k can be discovered for which $\mathrm{SD} \cup \mathrm{OBS} \cup \{\neg ab(c) | c \in mcs\} \vdash_k^{BCP} \bot$, which actually comes down to checking how the conflicting observations can be propagated through the part of the system that is the minimal conflict set. The behaviour of components that are not in mcs does not matter: they are not assumed to be working correctly and no conclusions can be drawn for them. For this, first the minimal conflict sets have to be found. The task of determining the minimal conflicts sets is of the same complexity as performing classical diagnostic reasoning; from the minimal conflict sets, the classically correct diagnoses can very easily be determined.

As a result, the upper bound k^* from the above theorem is not one that can be effectively exploited: even though we know it exists, determining its actual value is of the same complexity as performing (classical) diagnosis in the first place.

Of course, one expects that the observation-dependent upperbound k^* is sharper than k_{max}, the observation-independent upperbound. This is indeed the case:

Theorem 7 *For any problem (SD,COMP,OBS):*

$$k_{max}(SD) \geq k^*$$

The following table shows the values of k_{max} and k^* for all the examples from figure 1:

Example no.	k_{max}	k^*
Example 1	2	0
Example 2	2	1
Example 3	3	3
Example 4	3	2

Theorem 7 states that k_{max} is only a sufficient but not necessary upperbound. On the other hand, k_{max} is easily computable (linear in the size of the circuit) while the cost of computing k^* is of the same order as performing classical diagnosis in the first place. The figures from the above table show that the cheap upperbound k_{max} does not overshoot the optimal value by very much, at least on the small examples from figure 1.

5.5 BINARY CIRCUITS CAN BE DIAGNOSED WITH k=1

Systems consisting of the traditional one- or two-input gates such as AND, OR and NOT gates play a central role in most literature on formalising diagnosis. Theorem 8 states a surprising result about such systems consisting only of components with at most two inputs.

Theorem 8 ($k = 1$ suffices for diagnosing binary circuits)
For all electronic circuits consisting of components with no more than two inputs, Δ *is a* BCP_1 *diagnosis* iff Δ *is a classical diagnosis.*

Proof sketch: Even full classical entailment can only derive results about components with at least one known input or output signal. Choose such a component. Lemma 1 guarantees that for this component BCP_1 can derive all classically valid results (since $n = 3, m \geq 1$). Therefore, for this component, BCP_1 reasoning completely emulates classical reasoning. This argument can be repeated for every subsequent classical reasoning step. □

This is a quite surprising result, because apparently all diagnostic problems for this kind of systems can be solved with an entailment relation as simple as BCP_1, regardless of the size or structural complexity of the system.

5.6 RECOGNISING DIAGNOSTIC PROBLEMS WITH k=0

As can be seen from examples 2 and 3 from fig. 1, sometimes BCP_k is not able to even recognise the existence of a diagnostic problem. It is not able to derive inconsistency, and suggests the empty diagnosis (ie. no component is functioning incorrectly. The following theorem says that in certain cases, BCP_k is guaranteed to recognise a diagnostic problem if there is one.

Theorem 9 (recognising diagnostic problems at k=0)
For all electronic circuits for which all the inputs are known, Δ *is a* BCP_0 *diagnosis* iff Δ *is a classical diagnosis.*

Proof sketch: For recognising a diagnostic problem, we must prove

$$SD \cup \{\neg ab\_c | c \in COMP\} \cup \mathrm{OBS} \vdash_0^{BCP} \bot \quad (2)$$

For some components c, all inputs will be known (since all inputs to the circuit are assumed to be known). The description of components consists of clauses of the following form:

$$ab\_c \vee \neg in_1\_c\_v \vee ... \vee \neg \in_n \_c\_v \vee out\_c_v$$

Fro the literals given on the left-hand side of (2), this can be reduced to the literal out_c_v (because all inputs of c are known). This amounts to "removing component c from the circuit". This process can be repeated until no components are left. At that time, only literals are left, and inconsistency can be derived by BCP_0. □

The strength of the theorem is that it applies to systems of arbitrary complexity. The weakness of this theorem is that it only applies when all input signals are known. If n inputs values are unknown, then the theorem can be applied for all of the 2^n possible open combinations. If n is not too large, 2^n applications of unit resolution may still be more efficient then a single application of a general resolution procedure.

6 CONCLUSIONS

In this paper we have shown how approximate boolean constraint propagation can be used for approximate diagnostic reasoning.

We have taken the well-known definition of Reiter's consistency based diagnosis, and replaced the logical entailment in its definition with a sound but incomplete approximation of the logical entailment. Using such an approximated version less diagnostic problems are recognized, but when a diagnostic problem is recognized more diagnoses are found. We were able to characterise the cheapest versions of approximate BCP which allows single components and entire circuits to be diagnosed correctly. From these upperbounds surprisingly low values follow which are needed to correctly diagnose many of the typical examples from the literature. A particularly interesting property that we discovered is that the complexity level at which approximate diagnoses coincide with classical diagnoses is entirely determined by the nature of the individual components, and not by either the size or the structural complexity of the overall device.

The question remains whether k is a good indicator for the complexity of diagnostic problems. For traditional diagnosis of binary circuits, k is definitely not a good indicator: theorem 8 states that $k = 1$ already captures all complexity. This leaves $k = 0$ as the only approximating step. For more complex components with more input- and output gates, the value where k-diagnosis becomes classical is higher. In principle, makes it possible to approximate more gradually the classical case with increasing k. This would make k an attractive indicator of diagnostic complexity in the case of complex components. Whether or not this is the case cannot be established by the small examples that we have studied so far.

Related work In [tenTeije & vanHarmelen, 1996], we already studied approximate diagnosing by using the approximate entailment from Schaerf and Cadoli [Schaerf & Cadoli, 1995] in several definitions of diagnosis. Compared with this earlier work, one of the drawbacks of using BCP_k for approximation is that we are not able to give guidelines for a good choice for k. We can only say that one should start at $k = 0$ and must increase k until the upperbound has been reached. For the Cadoli/Schaerf approximation on the other hand, we were able to give strategies for good choices for the parameter that determines cost and quality of the approximation [tenTeije & vanHarmelen, 1997].

A second point in favour of the Schaerf/Cadoli approximations is the desired gradual behaviour of an approximate diagnostic method. For the Schaerf/Cadoli approximation, we have observed such gradual behaviour (see the examples in [tenTeije & vanHarmelen, 1996]), while for BCP_k this question is still open, and the initial evidence is discouraging. A further problem with BCP_k is that even the initial step in the approximating process ($k = 0$) is already quite strong (BCP_0 is complete for Horn theories), thereby removing much scope for approximating steps which were possible under the Schaerf/Cadoli approximation.

Other work in approximate diagnosis is done by Pos [Pos, 1993]. She takes a more algorithmic approach and applies the approach proposed in [Russel & Zilberstein, 1991] to diagnostic reasoning. The diagnosis task is decomposed in a number of subtasks (conflict generation, discrimination and conversion) and for each of these a number of anytime algorithms are proposed which should be chosen and scheduled. She discusses three methods for scheduling several anytime algorithms and applies these to diagnosis. This work is complementary to our approach, which deals with declarative specifications only.

Future work As mentioned above, it would be an attractive anytime behaviour if with every increase of k the quality of the diagnoses would increase a little bit. However, in our rather small experiments (both in size and in number), we often observe a behaviour that is the opposite of this: for small values of k the approximate diagnoses remained the same, only to suddenly jump to the entire set of classical diagnoses. The next step in our study of BCP_k is to do more experiments for testing under which conditions performing BCP_k diagnosis would result in better anytime behaviour. Another way of extending our experiments is to use more complicated behaviour models. The same should be done for investigating the tightness of the upperbound k_{max} on more realistic examples. In our examples we considered only models of the correct behaviour of components and therefore we are allowed to use the minimal consistency based diagnosis hypothesis [McIlraith, 1994]. A new subject of study is whether our results are depend on this minimal consistency based diagnosis hypothesis.

In [tenTeije & vanHarmelen, 1996], we showed that the use of a unsound but complete approximation gives useful results for performing diagnosis. Therefore another direction of further research is to look for an unsound but complete BCP approximation.

Acknowledgements

We are grateful to Wouter Teepe for providing an implementation of BCP_k, and to Rineke Verbrugge for supervising him during this work and for useful discussions in the early stages of this work.

References

[Boddy & Dean, 1989] M. Boddy and T. Dean. Solving time-dependent planning problems. In IJCAI–89.

[Bylander *et al.*, 1991] T. Bylander, D. Allemang, M. Tanner, and J. Josephson. The computational complexity of abduction. *Artificial Intelligence*, 49:25–60, 1991.

[Chang & Lee, 1973] C. Chang and R.C. Lee. *Symbolic Logic and Mechanical Theorem Provering.* Academic Press, London, 1973.

[Dalal & Yang, 1997] M. Dalal and L. Yang. Preliminary emperical results on anytime propositional reasoning (abstract). In *AAAI97 workshop*, 1997.

[Dalal, 1996] M. Dalal. Semantics of anytime family of reasoners. In ECAI'96, pages 360–364, 1996.

[Davis & Hamscher, 1988] R. Davis and W. C. Hamscher. Model-based reasoning: Troubleshooting. In H. E. Shrobe, editor, *Exploring Artificial Intelligence*, pages 297–346. Morgan Kaufmann, 1988.

[Dean & Boddy, 1988] T. Dean and M. Boddy. An analysis of time-dependent planning. In AAAI'88, pages 49–54, 1988.

[Green, 1969] C. Green. Application of theorem proving to problem solving. InIJCAI'69, pages 219–239, 1969.

[McAllester, 1990] D. McAllester. Truth maintenance. In AAAI'90, pages 1109–1116, 1990.

[McIlraith, 1994] S. McIlraith. Further contributuins to characterizing diagnosis. *Annals of Mathematics and AI*, 11(1–4), 1994. Special issues on Model-based diagnosis.

[Pos, 1993] A. Pos. Time-constrained model-based diagnosis. Technical report, University of Twente, department of computer science, 1993. Master's thesis.

[Reiter, 1987] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32:57–96, 1987.

[Rosenschein, 1981] S. Rosenschein. Plan synthesis: a logical perspective. In IJCAI'81, pages 331–337, 1981.

[Russel & Zilberstein, 1991] S. Russel and S. Zilberstein. Composing real-time systems. In IJCAI'91, pages 212–217, 1991.

[Schaerf & Cadoli, 1995] M. Schaerf and M. Cadoli. Tractable reasoning via approximation. *Artificial Intelligence*, 74(2):249–310, April 1995.

[tenTeije & vanHarmelen, 1996] A. ten Teije and F. van Harmelen. Computing approximate diagnoses by using approximate entailment. In KR'96, 1996. Morgan Kaufman.

[tenTeije & vanHarmelen, 1997] A. ten Teije and F. van Harmelen. Exploiting domain knowledge for approximate diagnosis. In IJCAI'97, pages 454–459, August 1997.

[vanHarmelen & tenTeije, 1995] F. van Harmelen and A. ten Teije. Approximations in diagnosis: motivations and techniques. In A. Levy and P. Nayak, editors, *Proceedings of SARA-95, Symposium on Abstraction, Reformulation, and Approximation*, pages 149–155, Aug 1995.

[Zilberstein & Russell, 1996] S. Zilberstein and S.J. Russell. Optimal composition of real-time systems. *Artificial Intelligence*, 82(1–2):181–213, 1996.